DESCRIPTION OF A TRANSIENT THERMAL ANALYSIS PROGRAM FOR USE WITH THE METHOD OF ZONES

by

Wilbur K. Sepetoski
Carol H. Sox
Peter F. Strong

Report

to

Jet Propulsion Laboratory

This work was performed for the Jet Propulsion Laboratory, California Institute of Technology sponsored by the National Aeronautics and Space Administration under Contract NAS7-100.

Report No. II
August 1963
C-65670

TABLE OF CONTENTS

DESCRIPTION OF A TRANSIENT THERMAL ANALYSIS PROGRAM FOR USE WITH THE METHOD OF ZONES

SUMMARY

26354

A program has been written (and from time to time modified) to solve numerically the differential equations which arise when the method of zones* is used to construct a mathematical model of the thermal behavior of a physical system. One particular feature of the program is that a number of checks are incorporated to improve the confidence in the input data and the results. Another feature is the use of an integration parameter, specified in the input data, to give control over the implicit integration procedure used.

A description of the problem and the checking procedures is first given in this report. Then the detailed specifications of the input data are discussed, and a sample problem presented.

The outputs, both on-line and off-line, are described. Finally, the full details of the program, including a catalogue of the storage contents, the FORTRAN coding, and the flow charts, are given as appendices.

author

GENERAL DESCRIPTION

The transient thermal analysis program, TTA, is a FORTRAN program designed to solve a system of ordinary differential equations of the form

$$\sum_j \alpha_{ij} \sigma T_j^4 + \sum_j \beta_{ij} T_j + \gamma_i + \delta_i \frac{dT_i}{dt} = 0 \qquad (1)$$

which arise in heat transfer problems involving conduction and radiation.

---

* "The Method of Zones for the Calculation of Temperature Distribution," by P. F. Strong and A. G. Emslie, Report to Jet Propulsion Laboratory, July, 1963.

In addition to solving the equations, the program is designed to carry out certain checking procedures and to print the results of these checks. The principal checks are a test for the conservation of energy in the complete system and a test for the consistency of the input data.

To test the conservation of energy, the quantity E_S is computed from the formula

$$E_S = \sum_i \delta_i T_i(0) - \sum_i \delta_i \; T_i(t) - \sum_i \int_0^t \gamma_i \; dt$$

$$- \sum_i \int_0^t \alpha_i \; \sigma \; T_i^{\,4} \; dt - \sum_i \int_0^t \beta_i \; T_i \; dt \qquad (2)$$

and printed along with the temperatures T_i as the calculation progresses. If the computation were perfectly accurate, E_S should be zero at all times. In practice, of course, one requires that E_S be small. Other quantities, such as the total energy stored in the system

$$E_p = \sum_i \delta_i \quad T_i(t) \qquad (3)$$

and the negative of the total energy into the system

$$E_{in} = \sum_i \int_0^t \gamma_i \; dt \quad , \qquad (4)$$

are also printed in order to provide numbers that can be compared with E_S. The quantity E_p also serves to indicate when a system has reached the end of a transient, and E_{in} is to be used to check that the input power is correctly computed.

To test the consistency of the input data, row and column sum checks are made on the matrices $\left\{\alpha_{ij}\right\}$ and $\left\{\beta_{ij}\right\}$. The row sums

$$S_R = \sum_j \alpha_{ij} - \alpha_i \tag{5}$$

$$S_C = \sum_j \beta_{ij} \tag{6}$$

are calculated for all values of i and tested to be small in magnitude. Any sums over tolerance are printed out. Also the column sums

$$_CS_R = \sum_i \pm \alpha_{ij} - \alpha_j \tag{7}$$

$$_CS_C = \sum_i \pm \beta_{ij} - \beta_j \tag{8}$$

are calculated for each value of j and tested to be small in magnitude. Any column sums over tolerance are printed. The signs in the summations of Equations 7 and 8 are the same as the signs of the diagonal elements α_{ii} or β_{ii}.

Finally, the average power into each zone, $\overline{\gamma_i}$, is calculated and printed. In the case of periodically varying power inputs, the average power over a period is part of the input data, so that $\overline{\gamma_i}$ can be automatically checked. The total average power input is also calculated and printed.

While most of the printing is naturally done off-line, a few quantities are printed on-line so that the calculation can be terminated right at the start if the input data are faulty. The quantities printed are the number of equations read into the machine, the number (and type) of sum checks that failed, the number of periodic power inputs with the wrong average level, and the total average power input.

It is to be noted that the method of solving the difference equations corresponding to the differential equations (1) requires that a diagonal element be in every row of the matrix $\{|\alpha_{ij}| + |\beta_{ij}|\}$ or if $\{\alpha_{ij}\}$ and $\{\beta_{ij}\}$ have only zeros in a certain row, that the corresponding δ_i be non-zero. The program checks to make sure that this requirement is satisfied and automatically omits the solution of the equations if the requirement is not satisfied.

A number of other checks on the format of the input data are made. These are all trivial and are best described by reference to the detailed flow charts.

The ordinary differential equations (1) can be solved only when starting temperatures $T_i(0)$ are given. In certain equations (i.e., the joining equations), however, the quantity δ_i is zero. In that case, the corresponding value of $T_i(0)$ must be calculated so that all the joining equations are satisfied initially. The program automatically carries out this calculation before proceeding with the main computation.

The fundamental units used in the program are determined by the units of the Stefan-Boltzmann constant and are power (watts), temperature (degrees Kelvin), and length (centimeters).

INPUT DATA

The input data consist mainly of a sequence of card images on magnetic tape. The data for several problems may be placed in succession on a single tape and followed by an end-of-file record. Under sense switch control, it is possible to read in certain information on cards via the on-line card reader. For the purpose of this description, however, we shall assume all data to be on cards as they might be before card-to-tape conversion.

A single problem consists of the following:

I. A heading card. This is an alphanumeric title to be printed on each sheet of output for identification and description.

II. A series of cards containing numerical information describing the differential equations to be solved. This series of cards is divided into sets, each of which describes a single equation. The format for all of these cards is the same, and in FORTRAN notation is I5, F10.3, F10.3, I5, I5. A decimal point is customarily used in the two F-type fields so as to permit full flexibility. The first three fields contain the information needed to solve the problem, and the last two fields contain identification numbers. The identification in the fourth field is the equation number and the identification in the fifth field is a serial number within the set of cards describing an equation. The information in the first three fields in a single set of cards is described below.

A. The first card contains the equation number i in the first field, the quantities α_i and β_i in the second and third fields, repectively, as required for use in Equation 2.

B. A series of cards comes next to give view areas, weighted conductances, and power inputs which determine α_{ij}, β_{ij}, and γ_i for a single value of i in Equation 1. These cards are of two types:

1. The first type of card contains the quantities j, α_{ij}, and β_{ij} in the first three fields. The indices j may be in any order and a single index may be repeated more than once if desired. If an index is repeated, the view areas and conductances given will be summed by the computer.

2. The second type of card is used when a power input is to be specified. A power input may be either constant or vary periodically. In the case of a constant power, a single card is needed: the first two fields are zero and the third field contains the <u>negative</u> of the applied power. In all equations which contain input powers, it is <u>required</u> that the diagonal elements α_{ii} and β_{ii} be positive; otherwise the conservation of energy check (Equation 2) will fail. In case the power varies periodically, the power must be tabulated as a function of time over the period

on a mesh which is fine enough to permit the use of the trapezoidal rule to integrate the power over time. The cards used for describing a periodic power are:

a. A card which contains a zero in the first field, the period in suitable units of time in the second field, and the average power in the third field. If the power is an input power, the third field must be negative. The average power given will be used as a check when the data are read into the machine.

b. A series of cards which tabulate the power as a function of time. The first field is zero, the second field contains the time, and the third field contains the power. The third field must be negative if the power is an input power. The first time given must be zero, and the last time must be equal to the specified period. In case a discontinuity in power occurs, the time at which it occurs must be repeated, and the two values of power given.

C. The last card of each equation contains the negative of the equation number in the first field, the thermal mass δ_i in the next field, and the starting temperature in the last field. The starting temperature will be interpreted as a first approximation to the solution of the joining equations if the thermal mass is zero, and will be used as a starting temperature for a zone if the thermal mass is not zero. The starting temperature may also be supplied in another way, and may be omitted in this card. This will be fully discussed later.

III. A blank card to terminate the data on the zones.

IV. One or more control cards. The control cards contain nine fields, whose format is, in FORTRAN notation, F5.2, E10.1, F10.3, I5, I5, F10.3, F10.3, I5, F5.2. Decimal points are customarily used in all F-type and E-type fields.

The first field, F5.2, contains the acceleration factor used in solving the systems of simultaneous equations and is generally taken to be in the range 1.2 to 1.6.

The second field, E10.1, contains a tolerance which determines when the iterations for solving the system of simultaneous equations are to be terminated.

The third field, F10.3, contains the time increment to be used in solving the differential equations.

The fourth and fifth fields, I5, contain integers which specify the frequency of off-line and on-line print-outs, respectively.

The sixth field, F10.3, contains the maximum value of time for which the problem is to be run before reading another control card.

The seventh field contains a starting temperature or zero. If a starting temperature is specified (and it must be on the first control card), this starting temperature will be used for all zones except for those which have already had a starting temperature specified under II-C above. If the starting temperature punched in the second or later control card is zero, the computation will continue with the most recent temperatures as starting values but with any new parameters (e.g., time interval, acceleration factor, etc.) as specified by the new control card. If on a second or later control card the starting temperature is not zero, the computation will be restarted from time zero with the new parameters specified by the control card.

The eighth field, I5, specifies the largest value of the index i. Since some indices may be omitted, the number in the eighth field will be greater than or equal to the number of equations to be solved.

The ninth field, F5.2, contains the integration parameter α.

V. The last card is a blank card which, when read in place of a normal control card, causes the computer to proceed to read in data for the next problem.

The cards just described are read in as card images from magnetic tape. Under control of sense switch three, however, control cards can be read in from the on-line card reader. A blank card read in this manner causes the computer to skip any control cards that remain on tape and go on to the next problem.

For the sake of illustration, we present the data for a simple problem. Imagine that we wish to calculate temperatures in three plates arranged as shown in Figure 1. Let us suppose that zone 2, with mean

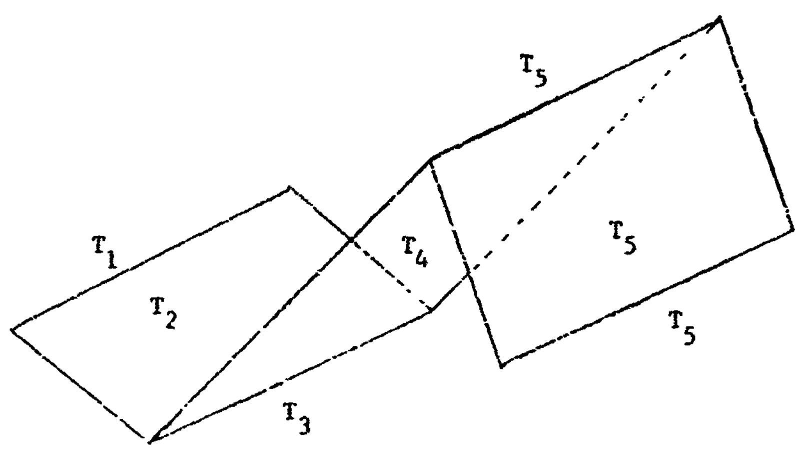


Figure 1. Three plates, thermally coupled by radiation and conduction.

temperature T_2, has a conductance of 1.5 watts per degree K, and is receiving a constant power input of 4 watts. We will also suppose that its area is 50 cm^2 and that the view area between zones 2 and 4 is 12 cm^2. The temperature T_1 is to be determined so that edge 1 is adiabatic. Zones 2 and 4 enjoy a common edge temperature T_3.

We will let zone 4 have a conductence of 0.7 watt per degree K, and have an area of 100 cm^2. The view area between zones 4 and 5 will be assumed to be 36 cm^2. We will assume a power input, P_4, to zone 4 to be a square wave of 10 watts lasting for 33 minutes, repeated every 110 minutes.

Zone 5 will be assumed to be a zone of constant temperature with $T_5 = 300°K$.

The thermal masses of the zones must also be specified. The usual units of thermal mass are joules per degree K, but in our work we have found it convenient to specify the time interval in minutes so we use watt-minutes per degree K instead. We will let the thermal masses of zones 2 and 4 be 3.0 and 8.0 watt-minutes per degree K, respectively.

The zone equations can now be written.

$$1.5(6T_2 - 4T_1 - 2T_3) = 0 \tag{9}$$

$$1.5(12T_2 - 6T_1 - 6T_3) + 100\sigma T_2^{\ 4} - 12\sigma T_4^{\ 4} - 4 + 3\,\frac{dT_2}{dt} = 0 \tag{10}$$

$$1.5(6T_2 - 4T_3 - 2T_1) + 0.7(6T_4 - 4T_3 - 2T_5) = 0 \tag{11}$$

$$0.7(12T_4 - 6T_3 - 6T_5) + 200\sigma T_4^{\ 4} - 12\sigma T_2^{\ 4} - 36\sigma T_5^{\ 4} - P_4 + 8\,\frac{dT_4}{dt} = 0 \tag{12}$$

$$\frac{dT_5}{dt} = 0 \tag{13}$$

We will let the starting temperatures of zones 2 and 4 be 280°K and 290°K, respectively.

The data for these equations, in the form required by the program, are given in Table I.

TABLE I

THE INPUT DATA FOR THE SAMPLE PROBLEM OF THREE PLATES

```
1   5    10   15   20   25   30   35   40   45   50   55   60   65
  THE EXAMPLE OF THREE PLATES
    1    0.0       0.0          1    1
    2    0.0       9.0          1    2
    1    0.0      -6.0          1    3
    3    0.0      -3.0          1    4
   -1    0.0       0.0          1   99
    2   88.0       0.0          2    1
    2  100.0      18.0          2    2
    1    0.0      -9.0          2    3
    3    0.0      -3.0          2    4
    0    0.0      -4.0          2    5
    4  -12.0       0.0          2    6
   -2    3.0     280.0          2   99
    3    0.0      -1.4          3    1
    2    0.0       9.0          3    2
    3    0.0      -6.0          3    3
    1    0.0      -3.0          3    4
    4    0.0  [illegible]       3    5
    3    0.0      -2.8          3    6
    5    0.0      -1.4          3    7
   -3    0.0       0.0          3   99
    4  188.0       4.2          4    1
    4  200.0       8.4          4    2
    3    0.0      -4.2          4    3
    5  -36.0      -4.2          4    4
    2  -12.0       0.0          4    5
    0  110.0      -3.0          4    6
    0    0.0     -10.0          4    7
    0   33.0     -10.0          4    8
    0   33.0       0.0          4    9
    0  110.0       0.0          4   10
   -4    8.0     290.0          4   99
    5  -36.0      -2.8          5    1
   -5    1.0     300.0          5   99
blank card
 1.6     1.0E-02   10.0         1   11   1100.0     300.0       5  0.6
blank card
```

The data on the first cards for the sets giving Equations 2, 3, 4, and 5 were arrived at by considering the net flux out of the system

$$Q_{out} = 0.7(6T_4 - 4T_5 - 2T_3) + 88\sigma T_2^4 + 185\sigma T_4^4 - 36\sigma T_5^4 \tag{14}$$

OUTPUT DATA

The bulk of the output consists of the time histories of the various temperatures. One page is printed off-line for each time interval specified on the control card. It consists of three lines of heading followed by pairs of index numbers and temperatures up to the highest number specified by the control card.

The first heading line is simply the alphanumeric heading information supplied on the first card of input. The second line gives the contents of the last control card read. The third line shows the values of time, E_p, E_S, and E_{in} as defined in Equations 3, 2, and 4, respectively, as well as the total number of iterations used in solving the systems of equations since the last on-line print.

Before the solution to the problem is printed, information about the input data is recorded. The off-line print consists of

1. The heading,

2. The total average power for each equation (unless zero),

3. The equation number and sum of the elements of any rows whose sum is over tolerance. As shown in Equations 5 and 6, S_R is used to denote the sum of radiation terms and S_C is sued to denote the sum of conduction terms.

4. A line of print which gives the number of equations read in, N_E, the number of row sum failures, N_R and N_C, for radiation and conduction,

respectively, the number N_p, of periodic powers with incorrect average level, and P_T, the total average power to the system.

5. The column number and the sum of the elements of any columns whose sum is over tolerance. As shown in Equations 7 and 8, ${}_C S_R$ is used to denote the sum of radiation terms and ${}_C S_C$ is used to denote the sum of conduction terms.

6. A line of print giving N_{CSC} and N_{CSR}, which are the numbers of column sum failures for conduction and radiation, respectively.

The on-line print-outs are as follows:

1. The heading for the problem,

2. Any card images with the wrong equation number in field 4,

3. The quantities N_E, N_R, N_C, N_P, and P_T as given in No. 4 above,

4. The numbers N_{CSC} and N_{CSR} as given in No. 6 above,

5. A print-out of equation numbers lacking diagonal elements,

6. A print-out if no integration parameter is supplied,

7. A print-out of all control cards as they are read in either from tape or card reader,

8. A print-out of any negative starting temperatures encountered,

9. A print-out of the residual after every fifty iterations during the solution of the joining equations for the starting temperatures,

10. The values of time, E_p, E_S, E_{in} and the number of iterations since the last on-line print of this type.

DESCRIPTION OF FLOW CHARTS

The flow charts are shown on pages C-1 to C-10. In drawing them up, we have departed slightly from conventional notation and have adhered closely to the FORTRAN notation. In particular, we have used the equal sign and the word IF as they are used in the FORTRAN language. The numbers attached to the boxes of the flow chart are the statement numbers used in the coding. The symbols "RIT" and "WOT" are short for "Read Input Tape" and "Write Output Tape", respectively. The symbol "SS" stands for "sense switch."

The portion of the program on pages C-1 and C-2 is to read in the heading and the numerical information on the equations to be solved. Row sums are computed and checked, and any over tolerance are printed out. In addition the average power for each zone is calculated (and checked in the case of periodic powers) and the total average power is calculated. For all card images read in, a check is made to see that the proper equation number is in field 4.

On page C-3 is shown the print-out of some check quantities just calculated and the method of calculation of column sums. Also shown is the test for the existence of some diagonal element (α_{ii}, β_{ii}, or δ_i).

On page C-4 is shown the check of the column sums, the print-out of the number of failures, the read-in of the control card image from tape (a test is made to insure that the integration parameter is supplied), the set-up of the initial temperatures (with a check against negative temperatures) and a call to subroutine FIFI. This subroutine, shown on page C-10, calculates f_{4i} and f_{1i} given by the relations

$$f_{4i} = \alpha \Delta t \, \alpha_{ii} \tag{15}$$

$$f_{ii} = \alpha \Delta t \, \beta_{ii} + \delta_i \tag{16}$$

where α is the integration parameter
Δt is the time increment.

The portion of the program on pages C-5 and C-6 is to calculate $E_p(0)$ and the initial values of temperatures whose derivatives do not explicitly appear in the given equations (i.e., the joining equations). Under control of SS5 the whole course of the solution may be printed off-line; in case convergence is not obtained, a new control card may be read in from the on-line card reader under control of SS3. In case a particular quartic cannot be solved, SS4 may be used to force the computer to stop the iteration and test SS3. Note that a blank card read in under this condition will terminate the attempt to solve the problem currently set up in the machine and cause the computer to read in the next problem for solution without having to read in the coding anew.

The program on page C-7 is mainly to calculate, at the beginning of each time step, the quantity

$$f_{01i} = (1-\alpha)\Delta t \sum_j \left[\alpha_{ij}\sigma T_j^{\,4}(t) + \beta_{ij}T_j(t)\right] + \int_t^{t+\Delta t} \gamma_i(t)dt - \delta_i T_i(t) \tag{17}$$

Two other quantities, used in the overall energy check, are also computed.

On page C-8 are shown the flow charts for the calculation of quantities used in the energy balance, the on-line print of time, E_S, E_p, etc., and the off-line print of headings and temperature data, which takes place just before the time is incremented.

The main iteration sequence to solve the equations at the new time starts after the time is incremented and is shown on the lower half of page C-8 (below box No. 99) and on page C-9.

Shown on the lower half of page C-8 is the test on the residual R to terminate the iteration and the calculation of E_p at the end of the time interval. Also shown is the calculation of

$$f_{02i} = \sum_{j \neq i} \beta_{ij} T_j(t+h) + \alpha_{ij} \sigma T_j^4(t+h) \tag{18}$$

The quantity f_{02i} is calculated with the latest approximations to $T_j(t+h)$. This calculation is the first step in the solution of the equations at the new time.

On page C-9 the calculation of the residual and the solution of the individual equations are shown. The uses of SS3 and SS4 are to read in a new control card as before.

Finally, on page C-10, are shown two simple subroutines named FIFI and GIGI. Subroutine FIFI is to compute f_{4i} and f_{1i} as noted earlier, and subroutine GIGI is to carry out an integration needed to compute

$$\int_t^{t+\Delta t} \gamma_i(t)\, dt.$$

APPENDIX A. CATALOGUE OF STORAGE CONTENTS

In order to facilitate a detailed study of the flow charts and coding, a brief catalogue of the contents of the named storage locations as they appear on the flow charts and a description of certain critical parts of the flow chart will now be given. An asterisk is used to label storage locations used for more than one purpose.

NE is the number of equations counted in the input.
*NR is the number of row sums (radiation) which are non-zero.
NC is the number of row sums (conduction) which are non-zero.
NPBAR is the number of periodic powers with incorrect averages.
K is an index used for proceeding through the input data.
WT is the total average input power for the problem.
XM is an array containing the thermal masses (δ_i).
J is an index.
I is an array containing the first field of input data.
A is an array containing the second field of input data.
B is an array containing the third field of input data.
JQ contains the fourth field of input data.
JQ1 contains the fifth field of input data.
I1 is the equation number.

The test on I(K) in box 5 is to sense the blank card at the end of the data.

SR is the row sum, radiation
SC is the row sum, conduction.
W is the input power for a single equation.

The test on I(K) in box 10 is to sense power inputs and the end of the equation.

TAU is the period of a periodic power.

WB is the specified average power.

WS is the average power computed for checking.

CSC is an array containing column sums (conduction).

CSR is an array containing column sums (radiation).

CON is a control, normally zero, but set equal to unity if any diagonal elements are missing.

*K1 is an index used to go through the input data to find the diagonal element.

SG is the sign of the equation, as determined by the sign of the diagonal element.

REL is the acceleration factor used in the relaxation process.

TOL is the tolerance to be satisfied in solving the equations.

DTI is the time increment, Δt.

KWF is the write frequency (i.e., for off-line printing).

KPF is the print frequency (i.e., for on-line printing).

TIM is the maximum time to which the solution is to be carried.

TB is the beginning temperature from the control card.

KSIZE is the maximum value of equation number I1.

AL is the integration parameter α.

The test on TB in box 82 is to determine whether the temperatures should be set to their initial values and the time reset to zero, or not.

The flow chart beginning with box 56, page C-4, and ending with the transfer to box 67, page C-7, is to set up the initial temperatures and set time equal to zero.

T is an array containing the temperatures.

TI is the time.

EZ is $E_p(0)$.

EN is the net change in energy in one time step; i.e.:

$$- \sum_i \int_{t-\Delta t}^{t} (\alpha_i \sigma T_i^4 + \beta_i T_i) dt - \sum_i \int_{t-\Delta t}^{t} \gamma_i dt$$

EIN is $\sum_i \int_0^t \gamma_i dt$.

EP is $E_p(t)$.

*NR is the number of iterations between on-line printouts.

T4 is an array containing σT_i^4.

KPC is a counter to control on-line printing.

R3 is a residual used in solving the joining equations.

L is an index for off-line printing.

KXC is a counter to control on-line printing during the solution of the joining equations.

GO is the constant term in a joining equation.

G1 is the coefficient of T_i in a joining equation.

G4 is the coefficient of σT_i^4 in a joining equation.

I2 is an index.

*T1 is an approximation to the root of the equation $g_o + g_1 T + g_4 \sigma T^4 = 0$.

σ is the Stefan-Boltzmann constant, 5.6686 watts $cm^{-2}(deg\ K)^{-4}$.

DT is the correction to T1 obtained by Newton's method, and is also used for the correction to T_i indicated in each iteration.

EIN2 is the integral $\sum_i \int_{t-\Delta t}^t \gamma_i dt$ at the present time t.

P2 is the integrand $\sum_i \alpha_i \sigma T_i^4 dt + \sum_i \beta_i T_i dt$ at the present time t.

FO1 is an array containing the quantities f_{01i} defined in Equation 17.

DTC is the quantity $(1-\alpha)\Delta t$.

TI1 is the remainder of TI modulo TAU.

*K1 is an index used to retain the location in storage of the beginning of the tabulation of a periodic power.

TI1 is the lower limit of a panel over which the input power is integrated by the trapezoidal rule.

TIU is the upper limit of a panel over which the input power is integrated by the trapezoidal rule.

WI is the integral of the input power over one panel (computed by subroutine GIGI).

EIN1 is the same as EIN2 except EIN1 is for the previous time $t-\Delta t$.

DT3 is one half the time increment used at the previous time $t-\Delta t$.

ES is E_S defined by Equation 2.

R is the residual computed to determine when the equations at one time are satisfied so that the calculation may proceed to the next time.

FO2 is the number f_{02i} defined in Equation 18.

P1 is the same as P2 except P1 is for the previous time interval.

DTA is the quantity $\alpha\Delta t$.

FO is the constant term in an equation, $\alpha\Delta t F_{02i} + f_{01i} = f_0$.

F1 is an array containing the coefficient of T_i in an equation.

F4 is an array containing the coefficients of σT_i^4 in an equation.

R1 is the quantity $f_0 + f_1 T_i + f_4 \sigma T_i^4$.

*T1 is an approximate solution to the equation R1 = 0.

R2 is the derivative of R1 with respect to T_i for use in Newton's method of solving quartic equations.

DT2 is $\frac{1}{2}\Delta t$.

TM is the arithmetic mean of TIL and TIU.

APPENDIX B: COPY OF THE CODING

```
*       ADL PFS  TRANSIENT THERMAL ANALYSIS   TTA-27   4TH SEPT 1963
*       XEQ
*       LIST
C       ADL PFS  TRANSIENT THERMAL ANALYSIS   TTA-27   4TH SEPT 1963
        DIMENSION H(12),XM(300),I(5500),A(5500),B(5500),CSC(300),CSR(300)
        DIMENSION  T4(300),F4(300),FI(300),FOI(300)
        COMMON  XM,I,A,B,DT2,F4,FI,TIL,TIU,WI,K,DTI,AL,DTA
        DIMENSION T(300)
 1000 FORMAT ( 12A6 )
 1001 FORMAT ( 1H1, 12A6)
 1002 FORMAT (I5,F10.3,F10.3,I5,I5)
 1003 FORMAT (1H0,I5,2F15.6,2I5,4H  EQ  I5)
 1004 FORMAT (5H0  EQ I5,6H  SR = F15.6)
 1005 FORMAT (1H0, 35X, 4H  EQ I5,6H  SC = F15.6)
 1006 FORMAT (1H0, 70X, 4H  EQ I5,6H   P = F15.6)
 1007 FORMAT (1H0, 40X, 4H  EQ I5,   7H  TAU =F15.6,8H  PBAR =F15.6)
 1008 FORMAT(10H0      NE =I5,7H   NR =I5,7H   NC =I5,7H   NP =I5,
     1   7H   PT =F15.6)
 1009 FORMAT (5H0  EQI4,18HDOES NOT CONTAIN T  I4)
 1010 FORMAT (6H0  COL I4, 7H  CSC =F15.6)
 1011 FORMAT (6H0  COL I4, 7H  CSR =F15.6)
 1012 FORMAT (     9H0  NCSC =I4, 9H   NCSR = I4)
 1013 FORMAT (F5.2,E10.1,F10.3,I5,I5,F10.3,F10.3,I5,F5.2)
 1014 FORMAT (6H0 TI = F9.2,7H   EP =E15.8,7H   ES =E12.5,7H  EIN =
 10141E12.5,7H   NI= I5)
 1015 FORMAT  ( 5H0REL=F5.2, 6H  TOL=E8.1, 5H  DT=F5.1, 5H  WF=I4,
     1 5H  PF=I4,7H  TIMX=F9.1,5H  TB=F7.1,7H  SIZE=I4,8H  ALPHA=F5.2)
 1016 FORMAT(1H0,I3,F7.2,I5,F7.2,I5,F7.2,I5,F7.2,I5,F7.2,
     1            I5,F7.2,I5,F7.2,I5,F7.2,I5,F7.2,I5,F7.2)
 1017 FORMAT(17H0  J. EQNS. RES = E 11.4)
    1 READ INPUT TAPE 9, 1000, (H(J), J=1,12)
      NE = 0
      NR = 0
      NC = 0
      NPBAR = 0
      K = 1
      WT = 0.
      DO 2 J = 1,300
    2 XM(J) = 0.
      WRITE OUTPUT TAPE 6, 1001, (H(J), J=1,12)
      PRINT 1001, (H(J), J=1,12)
    3 READ INPUT TAPE 9, 1002, I(K), A(K), B(K), JQ, JQ1
      I1 = I(K)
      IF ( I(K) - JQ ) 4, 5, 4
    4 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
    5 IF ( I(K) ) 7, 6, 7
    7 SR = -A(K)
```

```
      SC  = 0.
      W   = 0.
      K = K +1
      NE = NE +1
    8 READ INPUT TAPE 9, 1002, I(K), A(K), B(K),  JQ, JQ1
      IF ( I1 - JQ ) 9, 10, 9
    9 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
   10 IF ( I(K) ) 11, 12, 13
   11 XM(I1) = A(K)
       K = K+1
   15 IF (ABSF(SR) - 0.001) 17,17,16
   16 NR = NR +1
      WRITE OUTPUT TAPE 6, 1004, I1, SR
   17 IF (ABSF(SC) - 0.00001) 19,19,18
   18 NC = NC + 1
      WRITE OUTPUT TAPE 6, 1005, I1,  SC
   19 IF ( W )  20, 3, 20
   20 WRITE OUTPUT TAPE 6, 1006, I1, W
      WT = WT + W
      GO TO 3
   13 SR = SR + A(K)
      SC = SC + B(K)
   14 K = K + 1
      GO TO 8
   12 IF ( A(K) )  22, 21, 22
   21 W = W + B(K)
      GO TO 14
   22 TAU = A(K)
      WB   = B(K)
      WS   = 0.
      K    = K + 1
      READ INPUT TAPE 9, 1002, I(K), A(K), B(K), JQ, JQ1
      IF ( I1- JQ) 23, 24, 23
   23 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
   24 IF ( A(K))  25, 26, 25
   25 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
   26 K = K +1
      READ INPUT TAPE 9, 1002, I(K), A(K), B(K), JQ, JQ1
      IF( I1 - JQ) 27, 28, 27
   27 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
   28 IF ( A(K) - A(K-1) )  29, 30, 30
   29 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
   30 IF ( I(K) )  31, 32, 31
   32 WS = WS +((B(K) + B(K-1))* (A(K) - A(K-1))* 0.5)/TAU
      IF ( A(K) - TAU) 26, 33, 26
   33 IF ( ABSF ( WB -WS ) - 0.01) 35, 35, 34
   35 W = W + WS
      GO TO 14
   31 PRINT 1003, I(K), A(K), B(K), JQ, JQ1, I1
   34 WRITE OUTPUT TAPE 6, 1007, I1, TAU, WS
      NPBAR = NPBAR + 1
      GO TO 14
    6 PRINT 1008,NE,NR,NC,NPBAR,WT
      WRITE OUTPUT TAPE 6,1008, NE,NR,NC,NPBAR,WT
```

```
      DO 36 J=1, 300
      CSC  (J) = 0.
   36 CSR (J) = 0.
      NCSC = 0
      NCSR = 0
      CON  = 0
      K =1
   37 IF ( I(K)) 39, 38, 39
   39 J = I(K)
      CSR(J) = CSR(J) - A(K)
      CSC(J) = CSC(J) - B(K)
      K1= K + 1
   40 IF( I(K1) ) 116, 41, 41
  116 IF(A(K1))  117, 42, 117
   41 IF ( I(K1) - I(K)) 43, 44, 43
   43 K1 = K1 +1
      GO TO 40
   44 IF ( A(K1) + B( K1) ) 45, 46, 46
   45 SG = -1.
      GO TO 47
   46 SG = 1.
   47 K = K +1
      IF ( I(K) ) 48, 47, 49
   42 WRITE OUTPUT TAPE 6, 1009, I(K), I(K)
      PRINT 1009, I(K), I(K)
      CON = 1.
  117 K = K1
   48 K = K+1
      GO TO 37
   49 J = I(K)
      CSR(J) = CSR(J) + A(K)* SG
      CSC(J) = CSC(J) + B(K)* SG
      GO TO 47
   38 DO 50 J= 1, 300
      IF (ABSF(CSC(J)) - 0.00001) 50,50,51
   51 WRITE OUTPUT TAPE 6, 1010, J, CSC(J)
      NCSC = NCSC +1
   50 CONTINUE
      DO 52  J=1, 300
      IF (ABSF(CSR(J)) - 0.001) 52,52,53
   53 WRITE OUTPUT TAPE 6, 1011, J, CSR(J)
      NCSR = NCSR + 1
   52 CONTINUE
      WRITE OUTPUT TAPE 6, 1012, NCSC, NCSR
      PRINT 1012, NCSC, NCSR
   54 READ INPUT TAPE 9, 1013, REL, TOL, DTI, KWF, KPF, TIM,TB,KSIZE,AL
  501 PRINT1015, REL, TOL, DTI, KWF, KPF,TIM,TB, KSIZE, AL
      IF ( KSIZE ) 55, 1, 55
   55 IF(CON)  54, 82, 54
   56 K = 1
   57 IF ( I(K) ) 58, 59, 58
   58 I1 = I(K)
   61 K = K+1
      IF ( I(K) ) 62, 61, 61
```

```
   62 IF (B(K)) 151,64,63
  151 PRINT 1019,I1,B(K)
 1019 FORMAT ( 23HONEGATIVE TEMPERATURE T15, 2H =F7.2)
   63 T(I1) = ABSF(B(K))
      K= K+1
      GO TO 57
   64 T(I1) = ABSF(TB)
      K= K+1
      GO TO 57
   59 TI = 0.0
   60 EZ = 0.
      EN = C.
      EIN=0.
      IF (TB) 152,152,153
  152 PRINT 1020,TB
 1020 FORMAT ( 25HONEGATIVE TEMPERATURE TB=F7.2)
  153 CONTINUE
      DO 65  J=1, KSIZE
   65 EZ = EZ + T(J) * XM(J)
      CALL FIFI
      EP = EZ
      NR = 0
      TI = 0.
      KPC = KPF
      KWC = KWF
      KXC = 0
  123 K = 1
      KXC = KXC + 1
      IF(SENSE SWITCH 5) 149,150
  149 WRITE OUTPUT TAPE 6,1001,(H(J),J=1,12)
      WRITE OUTPUT TAPE 6,1018,REL,TOL,R3
 1018 FORMAT(1H0,  5H REL= F6.3,  5H TOL=E8.1,6H RESJ=E14.6)
      WRITE OUTPUT TAPE 6,1016,((L, T(L)),L=1,KSIZE)
  150 CONTINUE
      IF(KXC - 50)  147,148,148
  148 PRINT 1017, R3
      KXC = 0
  147 CONTINUE
      R3= 0.0
  124 IF(I(K)) 125,126,125
  125 I1 = I(K)
      G0 =  0.0
      G1 =  0.0
      G4 =  0.0
  127 K = K + 1
      IF(I(K)) 128,129,130
  128 IF(A(K)) 131,132,131
  131 K = K + 1
      GO TO 124
  129 IF(A(K)) 133,154,133
  154 G0 = G0 + B(K)
      GO TO 127
  133 TAU = A(K)
      G0 = G0 + B(K+1)
```

```
  134 K = K + 1
      IF(A(K) - TAU) 134,127,134
  130 IF(I(K) - I1) 135,136,135
  135 I2 = I(K)
      GO = A(K) * 5.6686E-12 *(T(I2)**4) + GO
      GO =  B(K) * T(I2) + GO
      GO TO 127
  136 G4 = A(K) + G4
      G1 = B(K) + G1
      GO TO 127
  126 IF(SENSE SWITCH 3) 137,138
  137 READ 1013, REL, TOL, DTI, KWF, KPF, TIM, TB, KSIZE, AL
      PRINT 1015, REL, TOL, DTI, KWF, KPF, TIM, TB, KSIZE, AL
      PAUSE
      IF(REL) 123,139,123
  139 READ INPUT TAPE 9,1013,REL
      IF(REL) 139,1,139
  138 IF(R3 - TOL) 67,67,123
  132 IF(G4) 140,141,140
  140 T1 = T(I1)
  142 DT = (GO + T1 *(G1 +5.6686E-12 *(G4*(T1**3))))/
     1(G1 + 22.6744E-12 *(G4 *(T1**3)))
      T1 = ABSF(T1 - DT)
      IF(ABSF(DT) - 0.04) 143,143,144
  143 DT = T1 - T(I1)
      R3 = R3 + ABSF(DT)
      T(I1) = T(I1) + REL * DT
      IF(T(I1)) 145,145,131
  145 T(I1) = ABSF(T1)
      GO TO 131
  144 IF(SENSE SWITCH 4) 146,142
  146 PAUSE
      GO TO 126
  141 T1 = - GO/G1
      GO TO 143
   67 EIN2 = 0.
      P2   = 0.
      K= 1
      DO 66 J= 1, KSIZE
   66 T4(J) = 5.6686E-12 *(T(J) ** 4)
      DO 68 J=1, KSIZE
   68 FO1(J) = 0.
   69 IF ( I(K) ) 71, 70, 71
   71 I1 = I(K)
      P2 = P2 + A(K) * T4(I1) + B(K) * T(I1)
   72 K = K+1
      IF ( I(K) ) 73, 74, 75
   73 K = K+1
      FO1(I1) = FO1(I1) - XM(I1) * T(I1)
      GO TO 69
   75 I2 = I(K)
      DTC = DTI - DTA
      IF ( A(K) )  76, 77, 76
   76 FO1(I1) = FO1(I1) + DTC * A(K) * T4(I2)
```

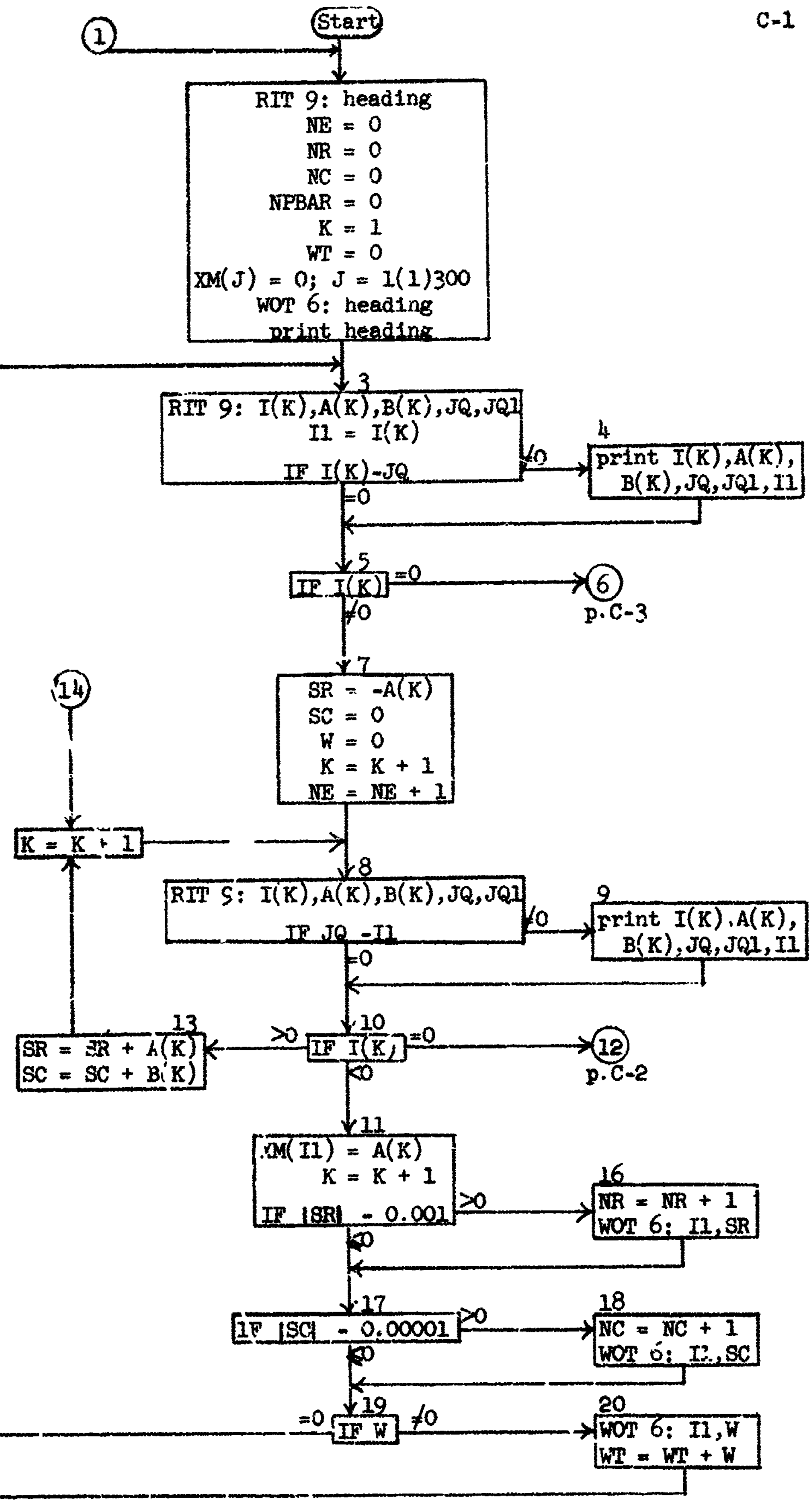
```
   77 IF ( B(K) ) 78, 72, 78
   78 FO1(I1) = FO1(I1) + DTC * B(K) * T(12)
      GO TO 72
   80 P1 = P2
      EIN1 = EIN2
      EP = 0.
      DT3 = DT2
      DO 81 J = 1, KSIZE
   81 EP = EP + T(J) * XM(J)
      IF (SENSE SWITCH 3 ) 79, 67
   79 READ 1013, REL, TOL, DTI, KWF, KPF, TIM, TB, KSIZE, AL
      PRINT 1015, REL, TOL, DTI, KWF, KPF, TIM, TB, KSIZE, AL
      PAUSE 77
      IF ( KSIZE ) 82, 83, 82
   82 IF ( TB ) 56, 84, 56
   84 CALL FIFI
      GO TO 67
   83 READ INPUT TAPE 9, 1013, REL
      IF(REL)  83, 1, 83
   74 IF(A(K)) 115,85,115
   85 FO1(I1) = FO1(I1) + DTI * E(K)
      EIN2 = EIN2 + DTI * B(K)
      GO TO 72
  115 TAU = A(K)
      TI1 = TI - FLOATF(XFIXF(TI/A(K))) * A(K)
      K1 = K+1
   86 K = K+1
      IF ( TI1 - A(K)) 87, 86, 86
   97 TIL = TI1
      K=K-1
   88 IF ( TI1+ DTI - A(K+1) )  89, 89, 90
   89 TIU = TI1+ DTI
      CALL GIGI
      FO1(I1) = FO1(I1) + WI
      EIN2 = EIN2 + WI
   91 K = K+1
      IF ( A(K) - TAU)  91, 72, 91
   90 TIU =  A(K+1)
      CALL GIGI
      FO1(I1) = FO1(I1) + WI
      EIN2 = EIN2 + WI
      K = K+1
      IF ( A(K) - TAU)  93, 92, 93
   92 TI1 = TI1 - TAU
      K = K1
   93 TIL = A(K)
      GO TO 88
   70 IF(TI) 94,95,94
   94 EN = EN - 2.*DT3*((1. - AL)*P1 + AL*P2) - EIN1
      EIN=EIN + EIN1
   95 IF ( KPC - KPF )  97, 96, 96
   96 ES = EZ + EN - EP
      PRINT 1014, TI, EP, ES, EIN, NR
      KPC = 0
```

```
      NR = 0
   97 IF ( KWC -KWF )  99, 98, 98
   98 WRITE OUTPUT TAPE11, 1001, (H(J), J=1,12)
      WRITE OUTPUT TAPE11,1015,REL,TOL,DTI,KWF,KPF,TIM,TB,KSIZE,AL
      ES = EZ + EN -EP
      WRITE OUTPUT TAPE11, 1014, TI, EP, ES , EIN, NR
      WRITE OUTPUT TAPE11, 1016,((L, T(L)), L=1,KSIZE)
      KWC =0
      IF(TI - TIM) 99,54,54
   99 TI = TI + DTI
      KWC = KWC +1
      KPC = KPC +1
  100 K = 1
      R = 0.
  101 IF ( I(K) ) 103, 102, 103
  102 IF ( ABSF(R) - TOL )  80, 80, 104
  104 NR = NR +1
      GO TO 100
  103 I1 = I(K)
      FO2 = 0.
  105 K = K+1
      IF ( I(K)) 106, 105, 107
  107 IF ( I(K) -I1) 108, 105, 108
  108 I2= I(K)
      FO2 = B(K)* T(I2) + A(K) * T4(I2) + FO2
      GO TO 105
  106 CONTINUE
  120 FO = FO2 * DTA + FO1(I1)
  122 IF(F4(I1)) 109,110,109
  109 R1 =  FO + F1(I1)* T(I1) + F4(I1) * T4(I1)
      T1 =  T(I1)
  111 R2 = 22.6744E-12 * F4(I1)*(T1 **3) + F1(I1)
      DT = R1/R2
      T1 = ABSF(T1 - DT)
      R1 = FO + T1 * ( F1(I1) + (T1**3) * 5.6686E-12 * F4(I1))
      IF(ABSF(DT) -0.04) 112,112,119
  110 R1 = FO + F1(I1)* T(I1)
      T1 = ABSF ( FO / F1(I1))
      IF (SENSE SWITCH 4) 118,112
  112 DT = T1 - T(I1)
      R = R + ABSF(DT)
      T(I1) = T(I1) + REL * DT
      IF ( T(I1)) 113, 113, 114
  113 T(I1) = T1
  114 T4(I1) = 5.6686E-12 * T(I1)**4
      K = K+1
      GO TO 101
  119 IF (SENSE SWITCH 4) 118, 111
  118 PAUSE
      IF(SENSE SWITCH 3) 79,54
      END (1,1,0,1,1)
```

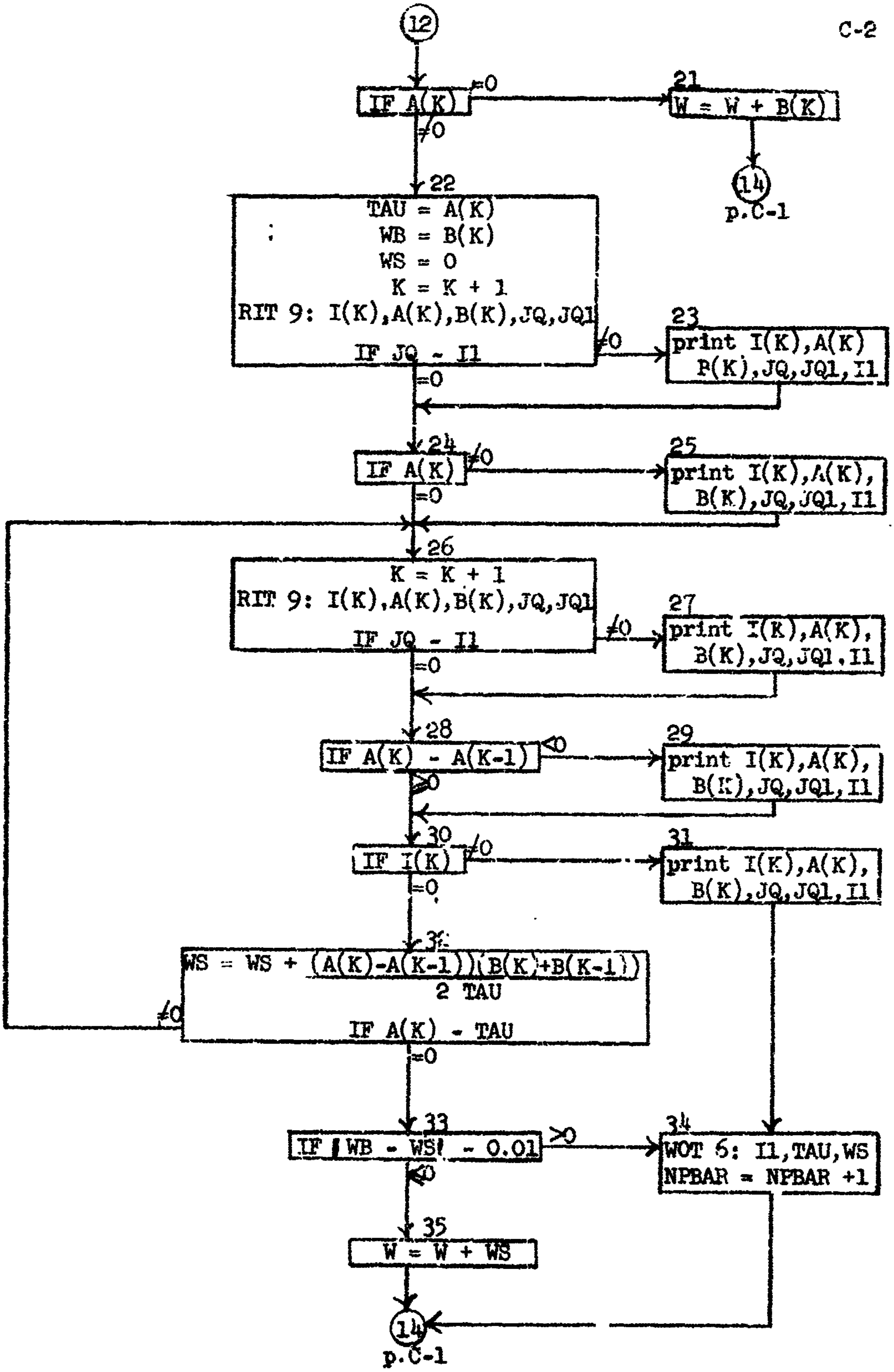
```
*      LIST
       SUBROUTINE  FIFI
       DIMENSION H(12),XM(300),I(5500),A(5500),B(5500),CSC(300),CSR(300)
       DIMENSION  T4(300),F4(300),F1(300),F01(300)
       COMMON  XM,I,A,B,DT2,F4,F1,TIL,TIU,WI,K,DTI,AL,DTA
       DT2 = 0.5 * DTI
       DTA = AL * DTI
       K = 1
     5 IF ( I(K) ) 1, 2, 1
     2 RETURN
     1 I1 = I(K)
       F4(I1) = 0.
       F1(I1) = 0.
     3 K = K+1
       IF ( I(K) ) 4, 6, 6
     4 F1(I1) = F1(I1) + XM(I1)
       K= K+1
       GO TO 5
     6 IF ( I(K) - I1 ) 3, 7, 3
     7 F4(I1) = F4(I1) + DTA * A(K)
       F1(I1) = F1(I1) + DTA * B(K)
       GO TO 3
       END (1,1,0,1,1)
```

```
*       LIST
        SUBROUTINE  GIGI
        DIMENSION H(12),XM(300),I(5500),A(5500),B(5500),CSC(300),CSR(300)
        DIMENSION  T4(300),F4(300),F1(300),FO1(300)
        COMMON  XM,I,A,B,DT2,F4,F1,TIL,TIU,WI,K,DTI,AL,DTA
        IF ( A(K) - A(K+1)) 1, 2, 1
      2 WI = 0.
        RETURN
      1 TM = 0.5 * ( TIL + TIU)
        WI = (( TIU-TIL) * ((A(K+1) - TM)*B(K) + (TM -A(K))*B(K+1))) /
       1(A(K+1) - A(K))
        RETURN
        END(1,1,0,1,1)
```

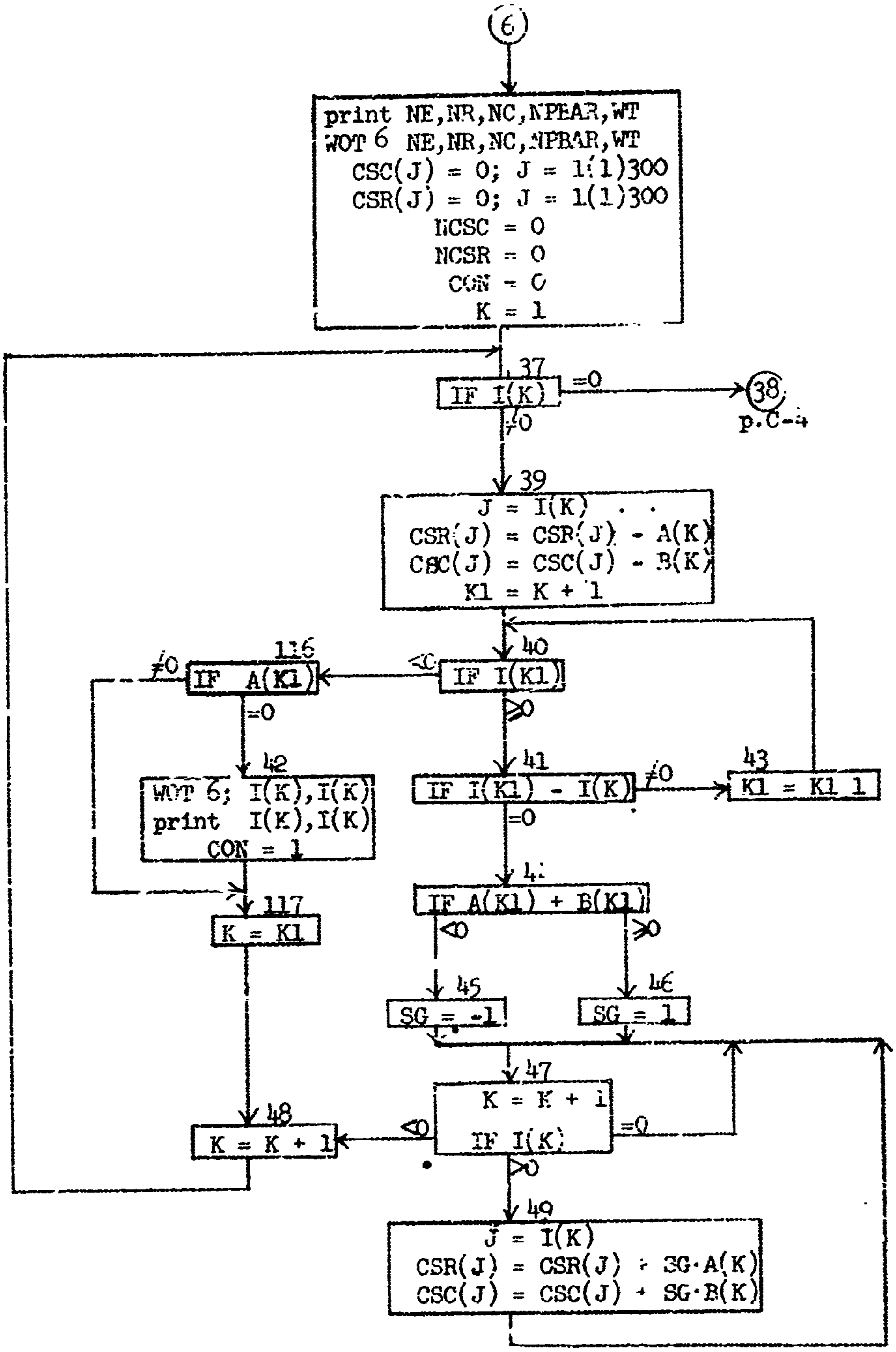
APPENDIX C: FLOW CHARTS

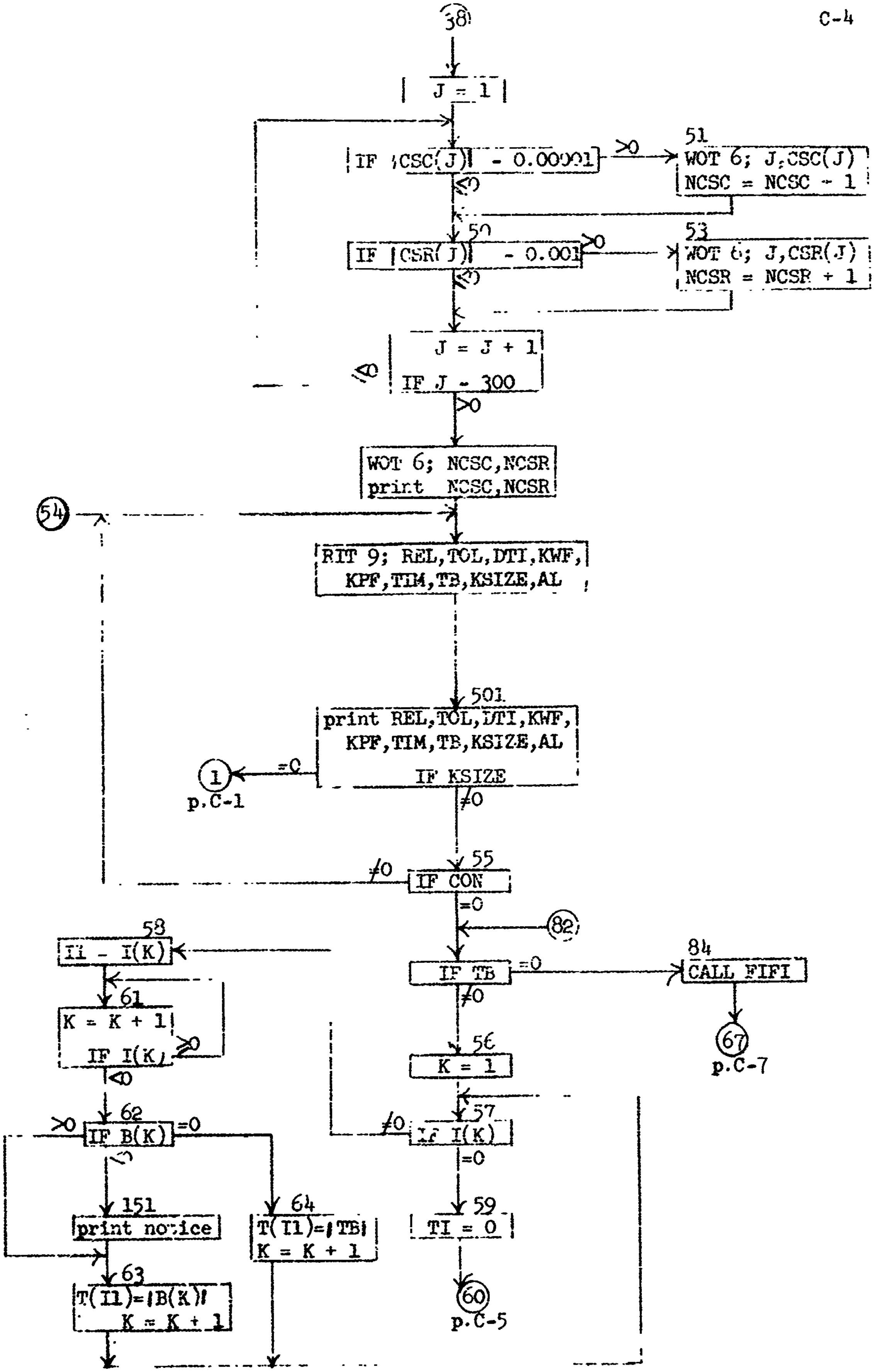


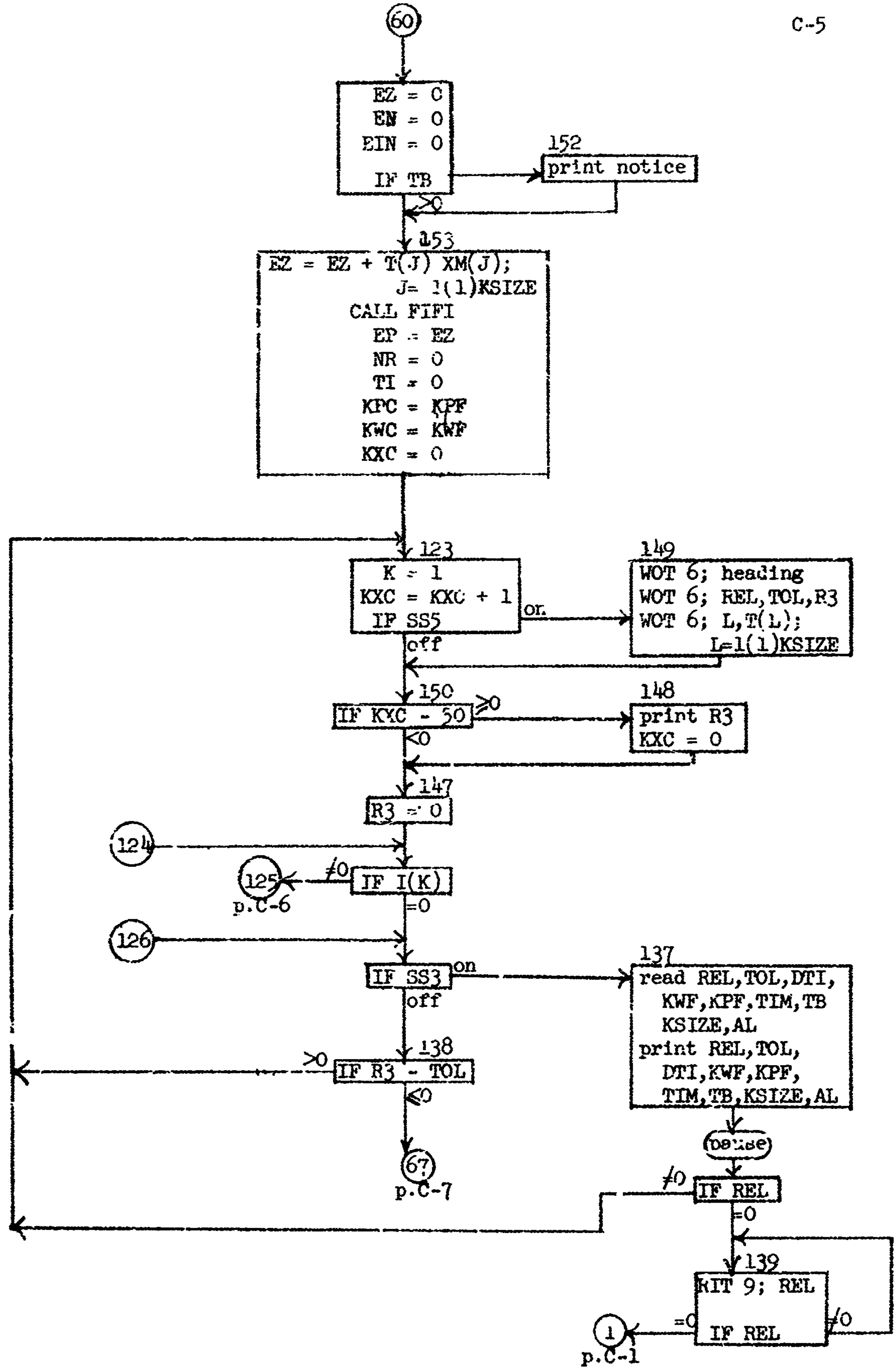
Start
1
RIT 9: heading
NE = 0
NR = 0
NC = 0
NPBAR = 0
K = 1
WT = 0
XM(J) = 0; J = 1(1)300
WOT 6: heading
print heading
3
RIT 9: I(K),A(K),B(K),JQ,JQ1
I1 = I(K)
IF I(K)-JQ
≠0
=0
4
print I(K),A(K),
B(K),JQ,JQ1,I1
5
IF I(K)
=0
≠0
6
p.C-3
7
SR = -A(K)
SC = 0
W = 0
K = K + 1
NE = NE + 1
14
K = K + 1
8
RIT 9: I(K),A(K),B(K),JQ,JQ1
IF JQ -I1
≠0
=0
9
print I(K),A(K),
B(K),JQ,JQ1,I1
10
IF I(K)
>0
=0
<0
12
p.C-2
13
SR = SR + A(K)
SC = SC + B(K)
11
XM(I1) = A(K)
K = K + 1
IF |SR| - 0.001
>0
≤0
16
NR = NR + 1
WOT 6: I1,SR
17
IF |SC| - 0.00001
>0
≤0
18
NC = NC + 1
WOT 6: I1,SC
19
IF W
=0
≠0
20
WOT 6: I1,W
WT = WT + W

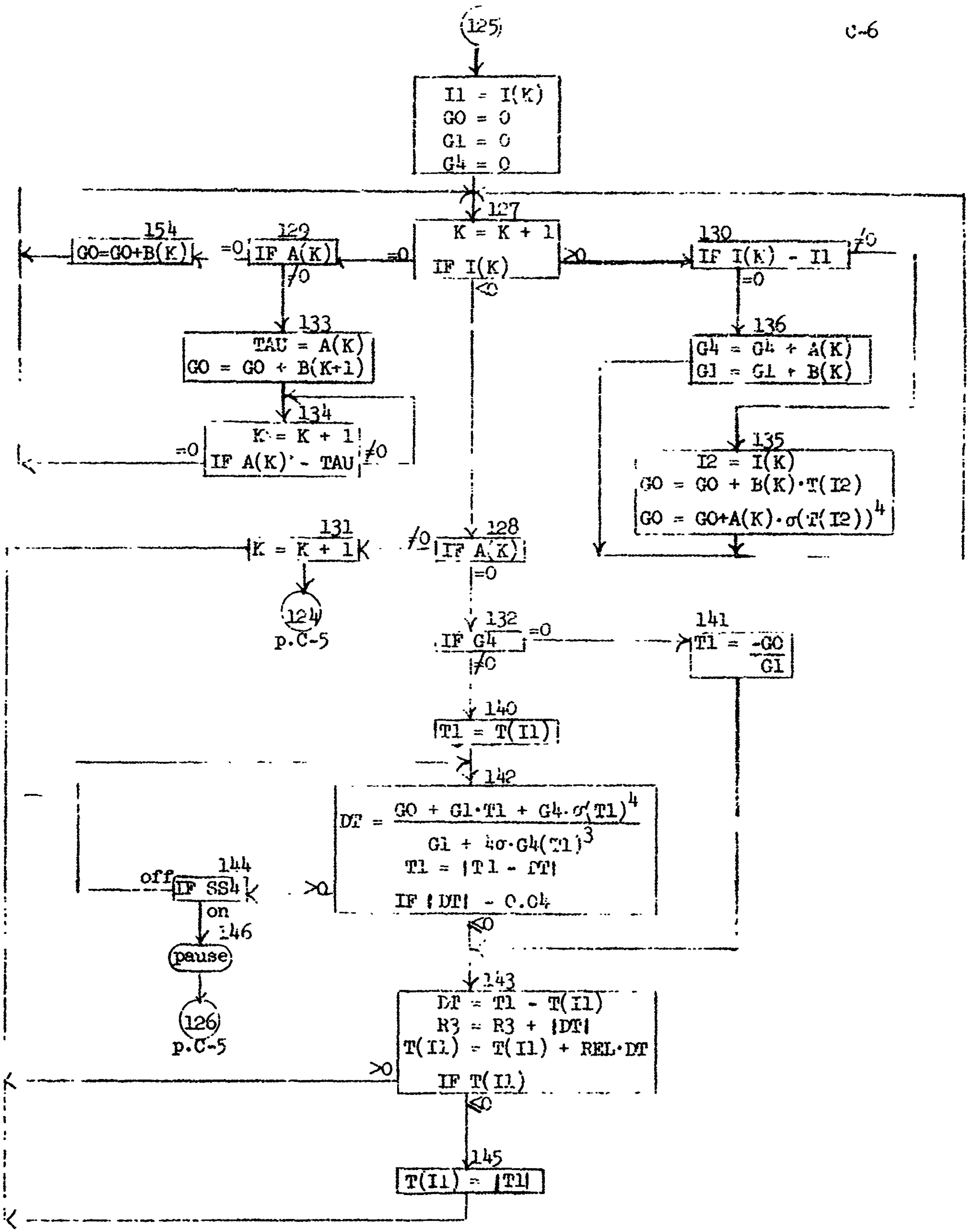


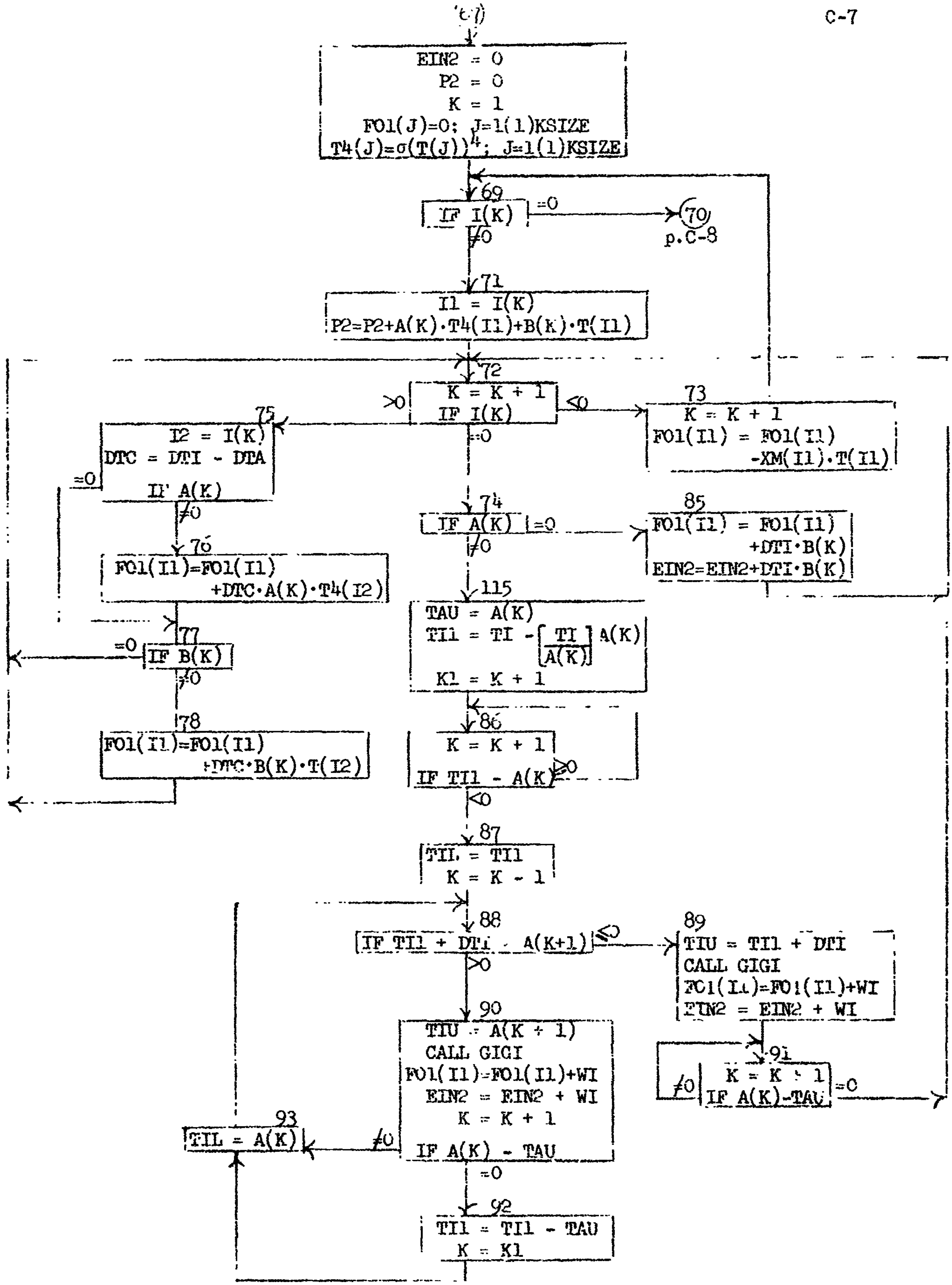
12
IF A(K)
=0
≠0
21
W = W + B(K)
14
p.C-1
22
TAU = A(K)
WB = B(K)
WS = 0
K = K + 1
RIT 9: I(K),A(K),B(K),JQ,JQ1
IF JQ - I1
≠0
=0
23
print I(K),A(K)
B(K),JQ,JQ1,I1
24
IF A(K)
≠0
=0
25
print I(K),A(K),
B(K),JQ,JQ1,I1
26
K = K + 1
RIT 9: I(K),A(K),B(K),JQ,JQ1
IF JQ - I1
≠0
=0
27
print I(K),A(K),
B(K),JQ,JQ1,I1
28
IF A(K) - A(K-1)
<0
≥0
29
print I(K),A(K),
B(K),JQ,JQ1,I1
30
IF I(K)
≠0
=0
31
print I(K),A(K),
B(K),JQ,JQ1,I1
32
WS = WS + (A(K)-A(K-1))(B(K)+B(K-1)) / 2 TAU
IF A(K) - TAU
≠0
=0
33
IF |WB - WS| - 0.01
>0
≤0
34
WOT 6: I1,TAU,WS
NPBAR = NPBAR +1
35
W = W + WS
14
p.C-1


6
print NE,NR,NC,NPBAR,WT
WOT 6 NE,NR,NC,NPBAR,WT
CSC(J) = 0; J = 1(1)300
CSR(J) = 0; J = 1(1)300
NCSC = 0
NCSR = 0
CON = 0
K = 1
37
IF I(K)
=0
38
p.C-4
≠0
39
J = I(K)
CSR(J) = CSR(J) - A(K)
CSC(J) = CSC(J) - B(K)
K1 = K + 1
40
IF I(K1)
<0
116
IF A(K1)
≠0
=0
≥0
42
WOT 6; I(K),I(K)
print I(K),I(K)
CON = 1
41
IF I(K1) - I(K)
≠0
43
K1 = K1 1
=0
117
K = K1
IF A(K1) + B(K1)
<0
≥0
45
SG = -1
46
SG = 1
47
K = K + 1
IF I(K)
=0
<0
48
K = K + 1
≥0
49
J = I(K)
CSR(J) = CSR(J) + SG·A(K)
CSC(J) = CSC(J) + SG·B(K)

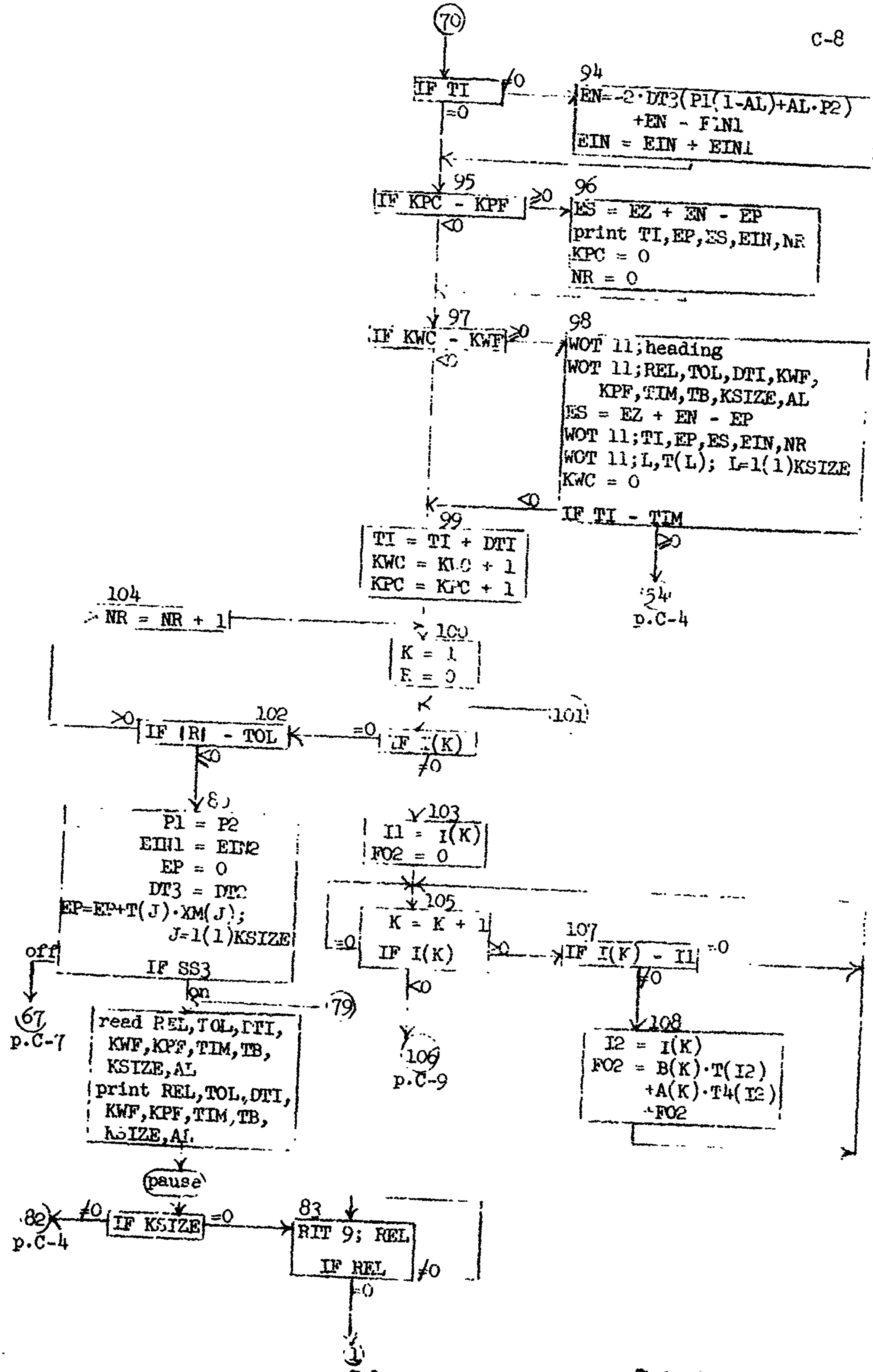

38
J = 1
IF |CSC(J)| - 0.00001
>0
51
WOT 6; J,CSC(J)
NCSC = NCSC + 1
≤0
50
IF |CSR(J)| - 0.001
>0
53
WOT 6; J,CSR(J)
NCSR = NCSR + 1
≤0
J = J + 1
IF J - 300
≤0
>0
WOT 6; NCSC,NCSR
print NCSC,NCSR
54
RTT 9; REL,TOL,DTI,KWF,
KPF,TIM,TB,KSIZE,AL
501
print REL,TOL,DTI,KWF,
KPF,TIM,TB,KSIZE,AL
IF KSIZE
=0
1
p.C-1
≠0
55
IF CON
≠0
=0
82
IF TB
=0
84
CALL FIFI
67
p.C-7
≠0
56
K = 1
57
IF I(K)
≠0
=0
59
TI = 0
60
p.C-5
58
I1 = I(K)
61
K = K + 1
IF I(K)
≥0
<0
62
IF B(K)
>0
=0
<0
151
print notice
63
T(I1)=|B(K)|
K = K + 1
64
T(I1)=|TB|
K = K + 1

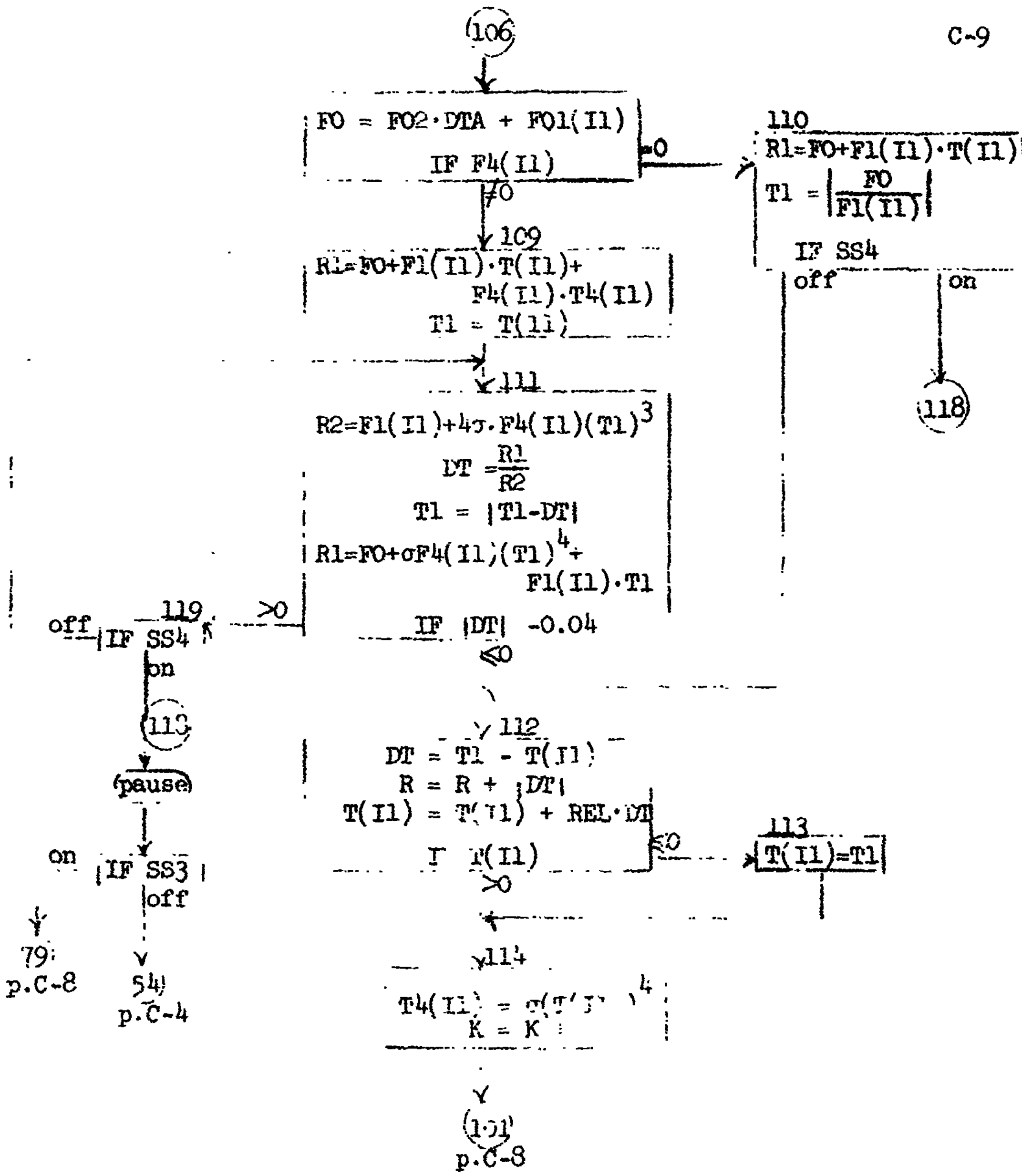

60
EZ = 0
EN = 0
EIN = 0
IF TB
152
print notice
>0
153
EZ = EZ + T(J) XM(J);
J= 1(1)KSIZE
CALL FIFI
EP = EZ
NR = 0
TI = 0
KPC = KPF
KWC = KWF
KXC = 0
123
K = 1
KXC = KXC + 1
IF SS5
on
off
149
WOT 6; heading
WOT 6; REL,TOL,R3
WOT 6; L,T(L);
L=1(1)KSIZE
150
IF KXC - 50
≥0
<0
148
print R3
KXC = 0
147
R3 = 0
124
125
p.C-6
≠0
IF I(K)
=0
126
IF SS3
on
off
137
read REL,TOL,DTI,
KWF,KPF,TIM,TB
KSIZE,AL
print REL,TOL,
DTI,KWF,KPF,
TIM,TB,KSIZE,AL
pause
138
>0
IF R3 - TOL
≤0
67
p.C-7
≠0
IF REL
=0
139
RIT 9; REL
IF REL
=0
≠0
1
p.C-1


125
I1 = I(K)
G0 = 0
G1 = 0
G4 = 0
127
K = K + 1
IF I(K)
154
G0=G0+B(K)
129
IF A(K)
130
IF I(K) - I1
133
TAU = A(K)
G0 = G0 + B(K+1)
134
K = K + 1
IF A(K) - TAU
136
G4 = G4 + A(K)
G1 = G1 + B(K)
135
I2 = I(K)
G0 = G0 + B(K)·T(I2)
G0 = G0+A(K)·σ(T(I2))^4
131
K = K + 1
128
IF A(K)
124
p.C-5
132
IF G4
141
T1 = -G0/G1
140
T1 = T(I1)
142
DT = (G0 + G1·T1 + G4·σ(T1)^4)/(G1 + 4σ·G4(T1)^3)
T1 = |T1 - DT|
IF |DT| - 0.04
144
IF SS4
off
on
146
pause
126
p.C-5
143
DT = T1 - T(I1)
R3 = R3 + |DT|
T(I1) = T(I1) + REL·DT
IF T(I1)
145
T(I1) = |T1|

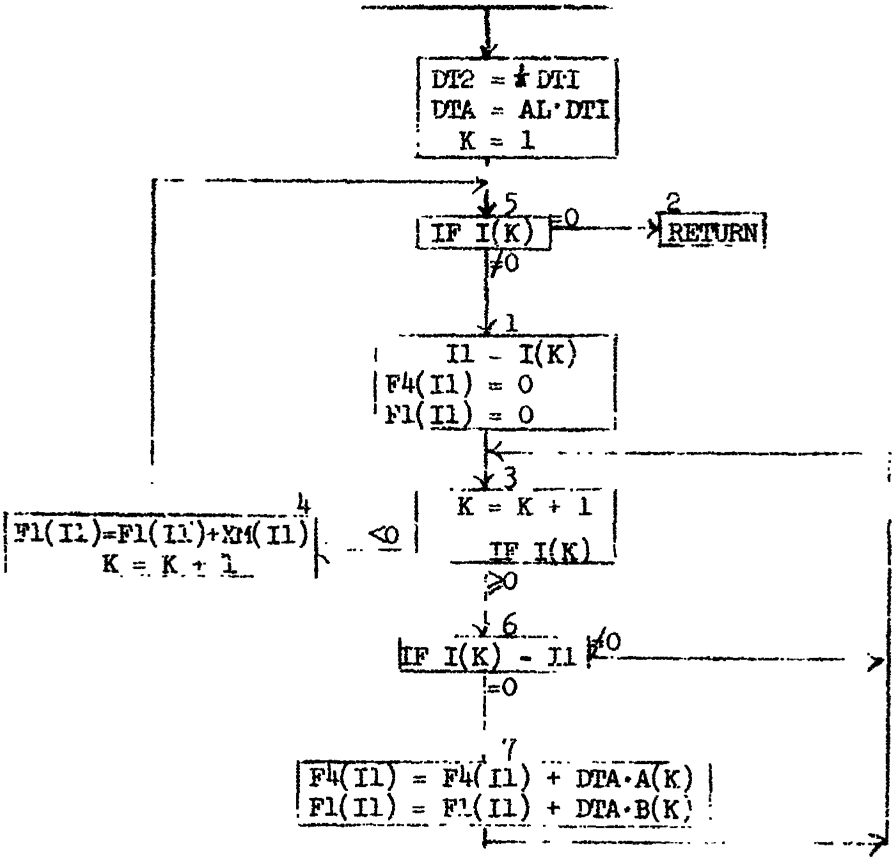


67
EIN2 = 0
P2 = 0
K = 1
FO1(J)=0; J=1(1)KSIZE
T4(J)=σ(T(J))^4; J=1(1)KSIZE
69
IF I(K)
=0
70
p.C-8
≠0
71
I1 = I(K)
P2=P2+A(K)·T4(I1)+B(K)·T(I1)
72
K = K + 1
IF I(K)
>0
<0
=0
73
K = K + 1
FO1(I1) = FO1(I1) -XM(I1)·T(I1)
75
I2 = I(K)
DTC = DTI - DTA
IF A(K)
=0
≠0
76
FO1(I1)=FO1(I1) +DTC·A(K)·T4(I2)
77
IF B(K)
=0
≠0
78
FO1(I1)=FO1(I1) +DTC·B(K)·T(I2)
74
IF A(K)
=0
≠0
85
FO1(I1) = FO1(I1) +DTI·B(K)
EIN2=EIN2+DTI·B(K)
115
TAU = A(K)
TI1 = TI - [TI / A(K)] A(K)
K1 = K + 1
86
K = K + 1
IF TI1 - A(K)
≥0
<0
87
TIL = TI1
K = K - 1
88
IF TI1 + DTI - A(K+1)
≤0
>0
89
TIU = TI1 + DTI
CALL GIGI
FO1(I1)=FO1(I1)+WI
EIN2 = EIN2 + WI
90
TIU = A(K + 1)
CALL GIGI
FO1(I1)=FO1(I1)+WI
EIN2 = EIN2 + WI
K = K + 1
IF A(K) - TAU
≠0
=0
91
K = K + 1
IF A(K)-TAU
≠0
=0
93
TIL = A(K)
92
TI1 = TI1 - TAU
K = K1


70
IF TI
≠0
=0
94
EN=-2·DT3(P1(1-AL)+AL·P2)
+EN - FIN1
EIN = EIN + EIN1
95
IF KPC - KPF
≥0
<0
96
ES = EZ + EN - EP
print TI,EP,ES,EIN,NR
KPC = 0
NR = 0
97
IF KWC - KWF
≥0
<0
98
WOT 11;heading
WOT 11;REL,TOL,DTI,KWF,
KPF,TIM,TB,KSIZE,AL
ES = EZ + EN - EP
WOT 11;TI,EP,ES,EIN,NR
WOT 11;L,T(L); L=1(1)KSIZE
KWC = 0
IF TI - TIM
<0
≥0
54
p.C-4
99
TI = TI + DTI
KWC = KWC + 1
KPC = KPC + 1
104
NR = NR + 1
100
K = 1
R = 0
101
102
IF |R| - TOL
>0
<0
=0
IF I(K)
≠0
80
P1 = P2
EIN1 = EIN2
EP = 0
DT3 = DT2
EP=EP+T(J)·XM(J);
J=1(1)KSIZE
IF SS3
off
on
67
p.C-7
79
103
I1 = I(K)
FO2 = 0
105
K = K + 1
IF I(K)
=0
>0
<0
106
p.C-9
107
IF I(K) - I1
=0
≠0
108
I2 = I(K)
FO2 = B(K)·T(I2)
+A(K)·T4(I2)
+FO2
read REL,TOL,DTI,
KWF,KPF,TIM,TB,
KSIZE,AL
print REL,TOL,DTI,
KWF,KPF,TIM,TB,
KSIZE,AL
pause
82
p.C-4
IF KSIZE
≠0
=0
83
RIT 9; REL
IF REL
≠0
=0
1


106
FO = FO2·DTA + FO1(I1)
IF F4(I1)
=0
≠0
109
R1=FO+F1(I1)·T(I1)+ F4(I1)·T4(I1)
T1 = T(I1)
110
R1=FO+F1(I1)·T(I1)
T1 = |FO / F1(I1)|
IF SS4
off
on
118
111
R2=F1(I1)+4σ·F4(I1)(T1)^3
DT = R1/R2
T1 = |T1-DT|
R1=FO+σF4(I1)(T1)^4+ F1(I1)·T1
IF |DT| -0.04
>0
≤0
119
off
IF SS4
on
pause
on
IF SS3
off
79
p.C-8
54
p.C-4
112
DT = T1 - T(I1)
R = R + |DT|
T(I1) = T(I1) + REL·DT
≤0
>0
113
T(I1)=T1
114
K = K
p.C-8

SUBROUTINE FIFI



SUBROUTINE GICI